Predicting the Risk of Heart Disease in a Person Based on their Biomedical Data

By Delisha Manuel

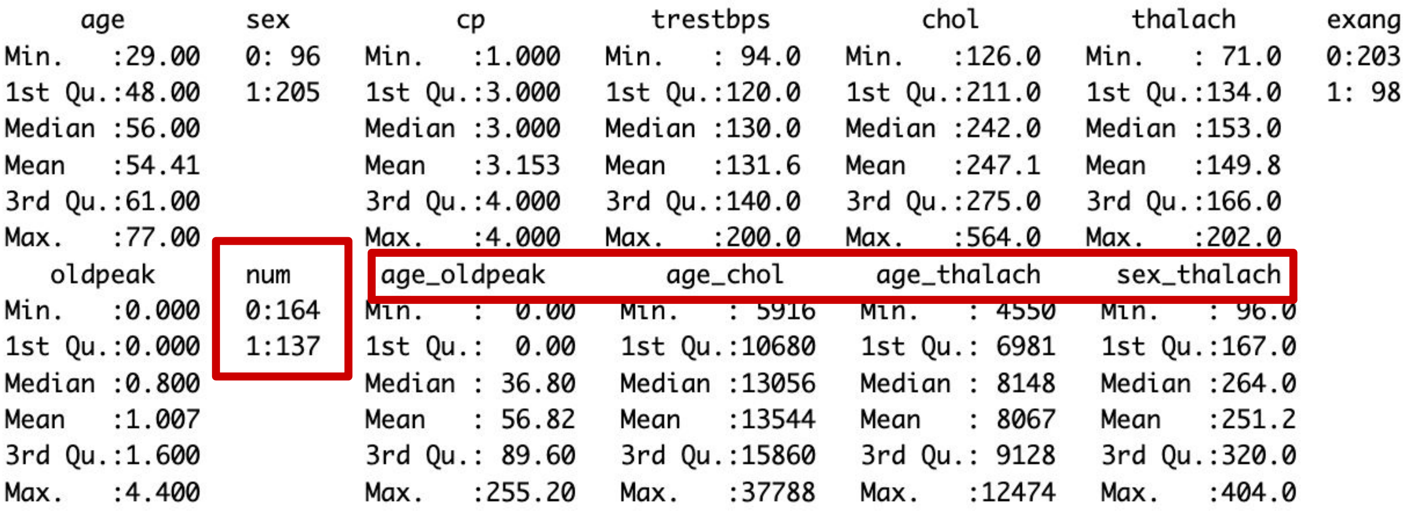
About My Project: Topic and Motivation

I am investigating the relationship between CP (presence of chest pain), trestbps (resting blood pressure), chol (serum cholesterol), fbs (fasting blood sugar), exang (exercise induced angina), oldpeak (ST depression induced by exercise relative to rest), thalach (max heart rate) age, and sex, and the probability that a patient has heart disease using a random forest model.

I chose this project because heart disease is one of the leading causes of death worldwide, and by predicting the probability that a patient has heart disease, we can choose which patients to prioritize for further diagnostic testing.

Introducing the Data

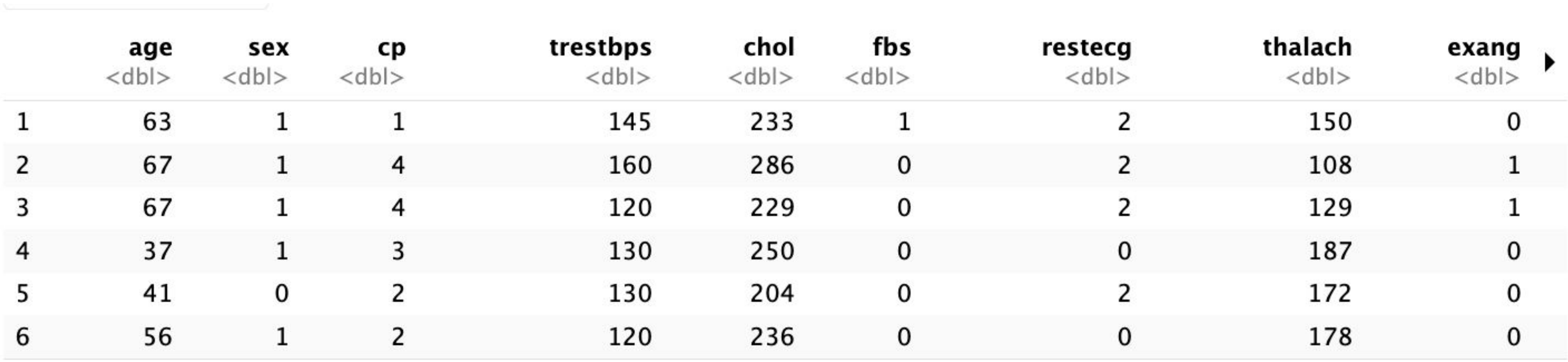
My dataset has 303 observations (rows) and 13 features. Although this does seem like a small dataset, it has been used extensively in academic research for heart disease prediction tasks. I also chose it because the data is publicly available and anonymized, so it does not raise any ethical concerns.



```
      age          sex          cp            trestbps          chol          thalach        exang
 Min.   :29.00   0: 96   Min.   :1.000   Min.   : 94.0   Min.   :126.0   Min.   : 71.0   0:203
 1st Qu.:48.00   1:205   1st Qu.:3.000   1st Qu.:120.0   1st Qu.:211.0   1st Qu.:134.0   1: 98
 Median :56.00           Median :3.000   Median :130.0   Median :242.0   Median :153.0
 Mean   :54.41           Mean   :3.153   Mean   :131.6   Mean   :247.1   Mean   :149.8
 3rd Qu.:61.00           3rd Qu.:4.000   3rd Qu.:140.0   3rd Qu.:275.0   3rd Qu.:166.0
 Max.   :77.00           Max.   :4.000   Max.   :200.0   Max.   :564.0   Max.   :202.0
    oldpeak         num      age_oldpeak         age_chol        age_thalach      sex_thalach
 Min.   :0.000   0:164   Min.   :  0.00   Min.   : 5916   Min.   : 4550   Min.   : 96.0
 1st Qu.:0.000   1:137   1st Qu.:  0.00   1st Qu.:10680   1st Qu.: 6981   1st Qu.:167.0
 Median :0.800           Median : 36.80   Median :13056   Median : 8148   Median :264.0
 Mean   :1.007           Mean   : 56.82   Mean   :13544   Mean   : 8067   Mean   :251.2
 3rd Qu.:1.600           3rd Qu.: 89.60   3rd Qu.:15860   3rd Qu.: 9128   3rd Qu.:320.0
 Max.   :4.400           Max.   :255.20   Max.   :37788   Max.   :12474   Max.   :404.0
```

Introducing the Data: Data Pre-Processing

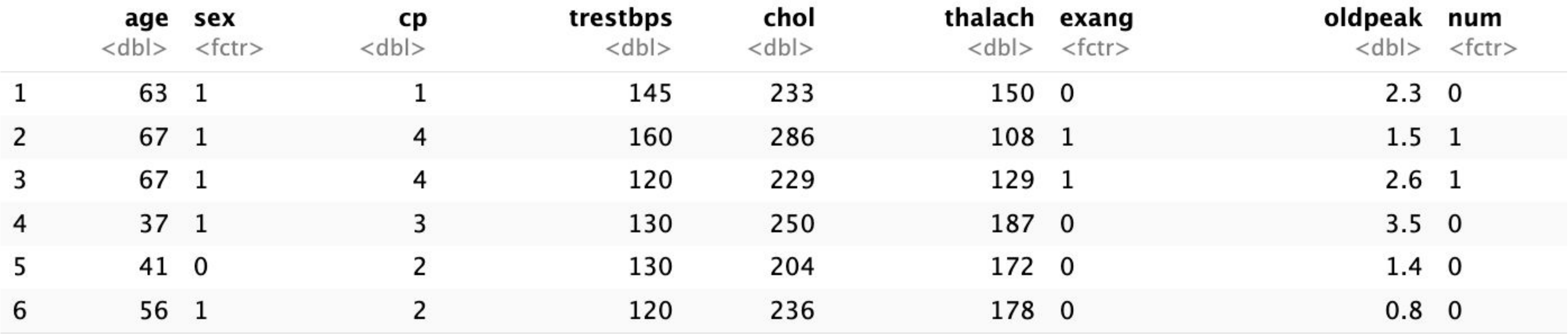
The na values were represented by a "?" (data type string), so when importing the dataset, I told R to convert all "?" strings into proper NA values. The column names were also not efficiently interpretable (just V1, V2, ...), so I renamed them to the actual attribute names.



	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	63	1	1	145	233	1	2	150	0
2	67	1	4	160	286	0	2	108	1
3	67	1	4	120	229	0	2	129	1
4	37	1	3	130	250	0	0	187	0
5	41	0	2	130	204	0	2	172	0
6	56	1	2	120	236	0	0	178	0

Introducing the Data: Data Pre-Processing

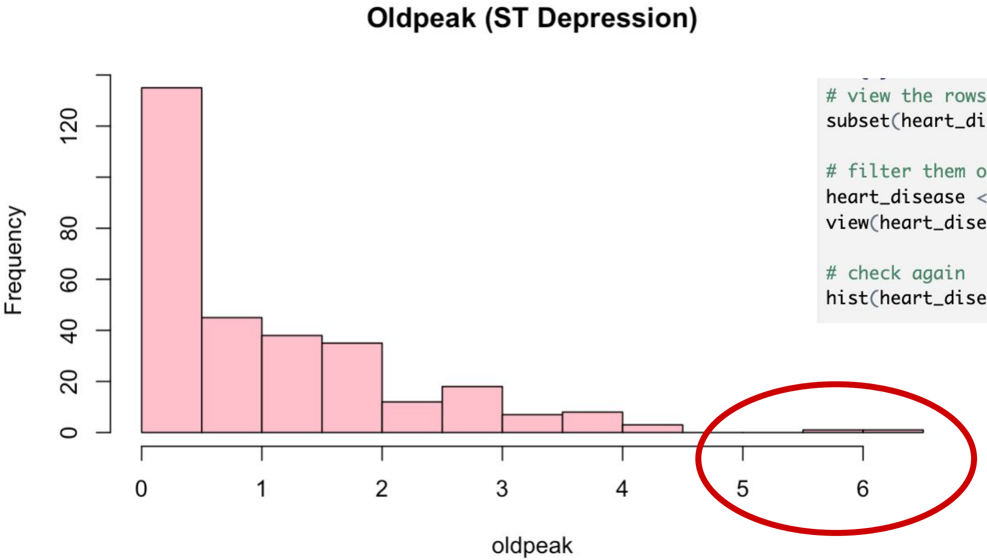
The features sex and exang are stored as doubles when they should be factors (categorical variables). The target variable, num, should also be a categorical value (0 or 1). Looking through documentation, I found that the numbers 0-4 indicate the severity of the heart disease, where 0 is no heart disease and 1-4 is having heart disease.

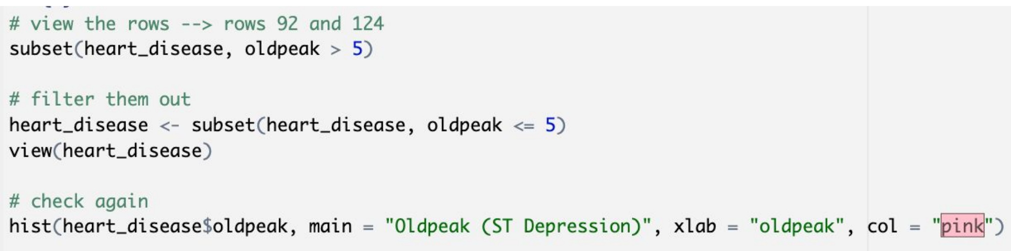


	age	sex	cp	trestbps	chol	thalach	exang	oldpeak	num
	<dbl>	<fctr>	<dbl>	<dbl>	<dbl>	<dbl>	<fctr>	<dbl>	<fctr>
1	63	1	1	145	233	150	0	2.3	0
2	67	1	4	160	286	108	1	1.5	1
3	67	1	4	120	229	129	1	2.6	1
4	37	1	3	130	250	187	0	3.5	0
5	41	0	2	130	204	172	0	1.4	0
6	56	1	2	120	236	178	0	0.8	0

Introducing the Data: Data Pre-Processing

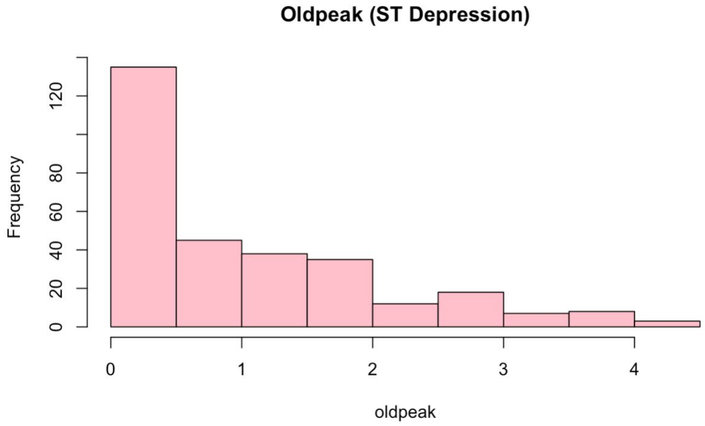
Based on the summary, I could tell there was an outlier in oldpeak (6.20), so I looked at a histogram to see where this is happening. Both the outliers were greater than 5, so I used that to look at and filter them out.

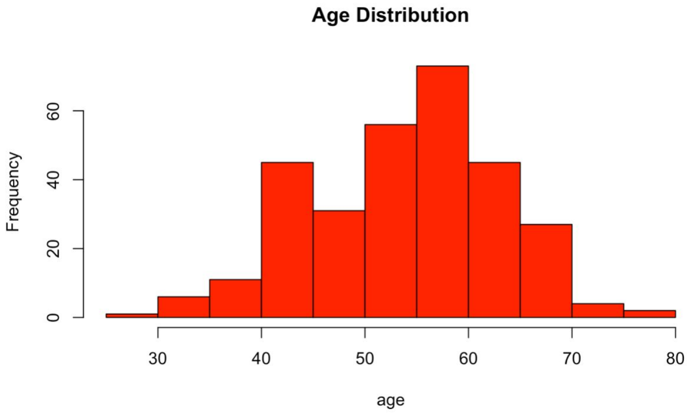


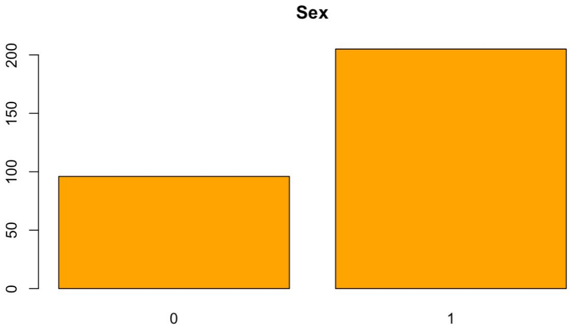


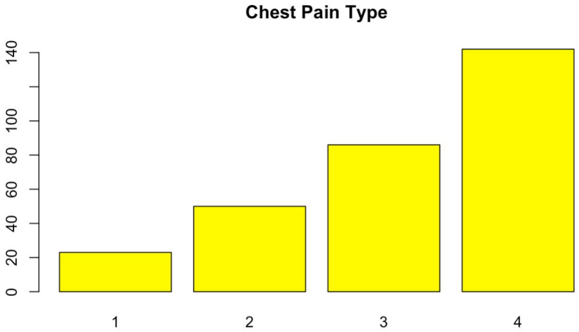
```
# view the rows --> rows 92 and 124
subset(heart_disease, oldpeak > 5)

# filter them out
heart_disease <- subset(heart_disease, oldpeak <= 5)
view(heart_disease)

# check again
hist(heart_disease$oldpeak, main = "Oldpeak (ST Depression)", xlab = "oldpeak", col = "pink")
```

Exploratory Data Analysis: Univariate Analysis

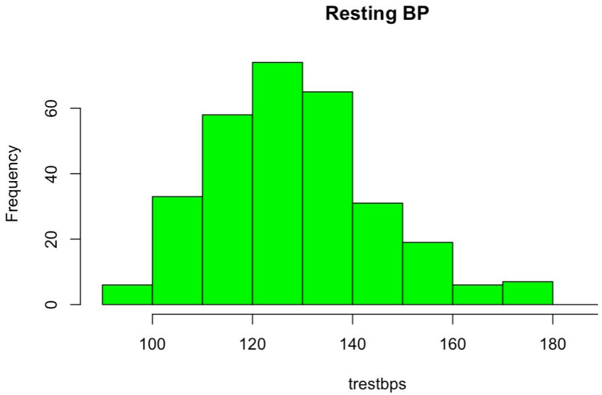


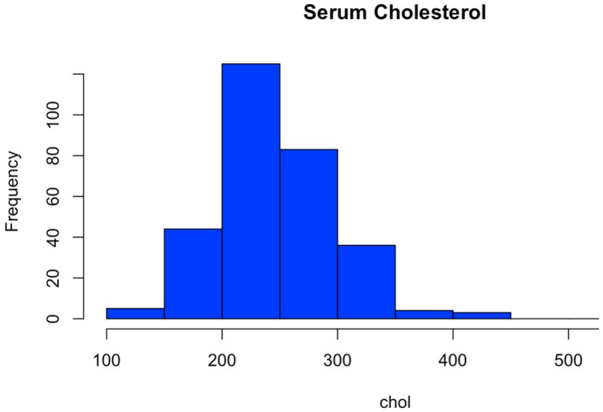


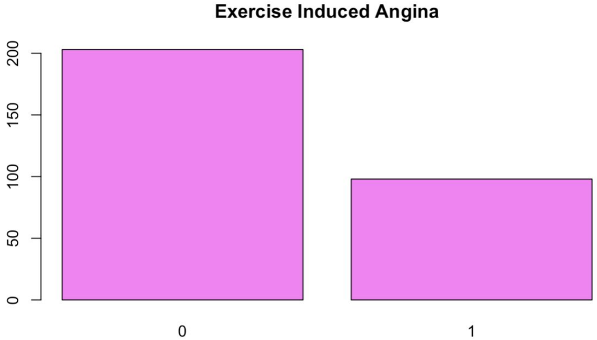


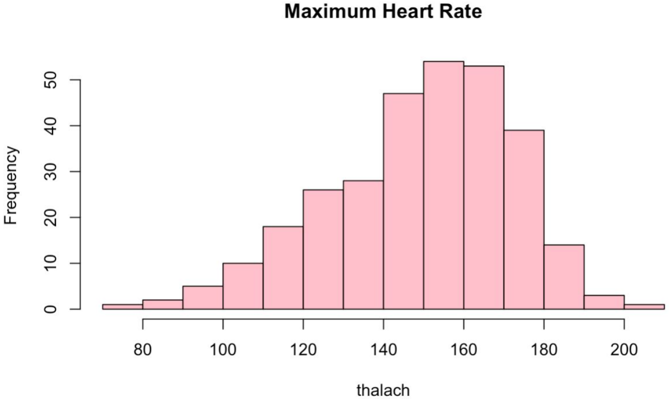


Exploratory Data Analysis: Univariate Analysis

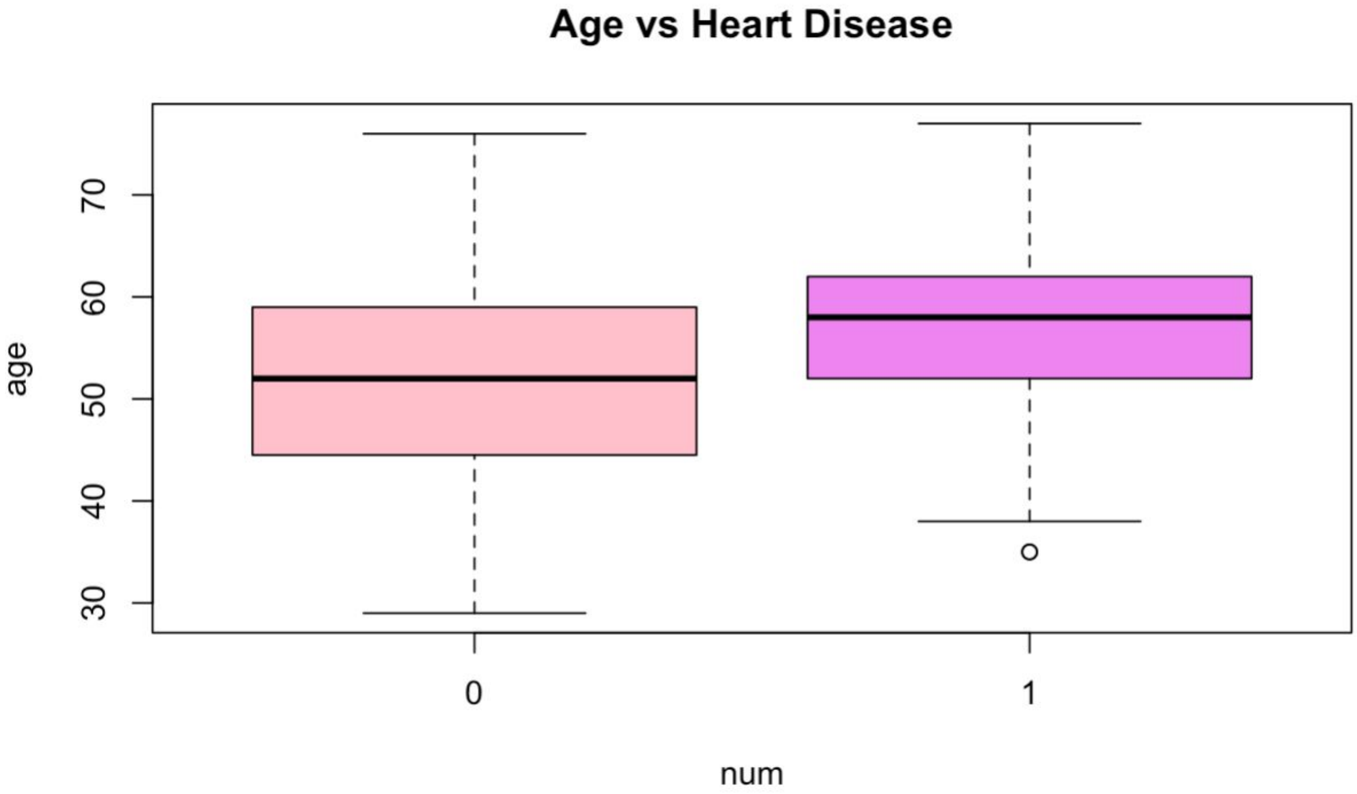

Resting BP


Serum Cholesterol

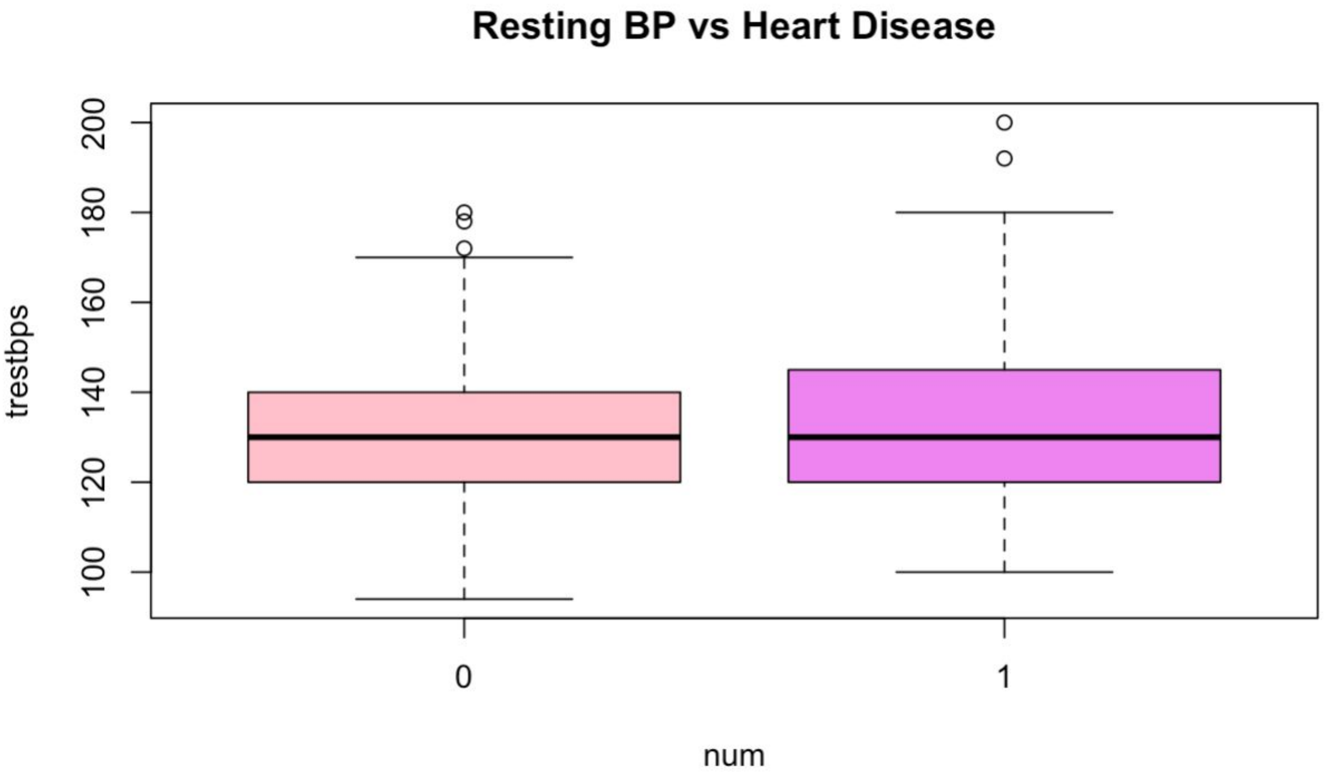

Exercise Induced Angina


Maximum Heart Rate

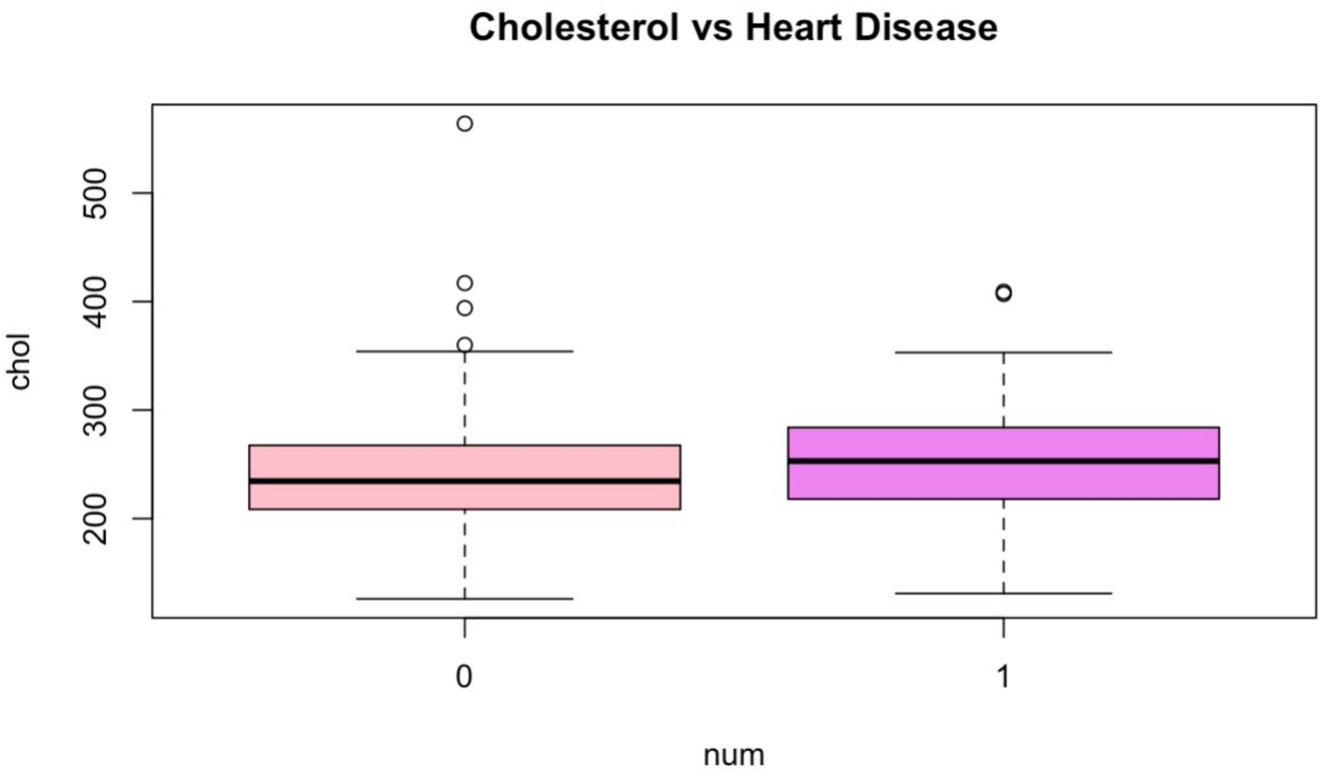
Exploratory Data Analysis: Multivariate Analysis



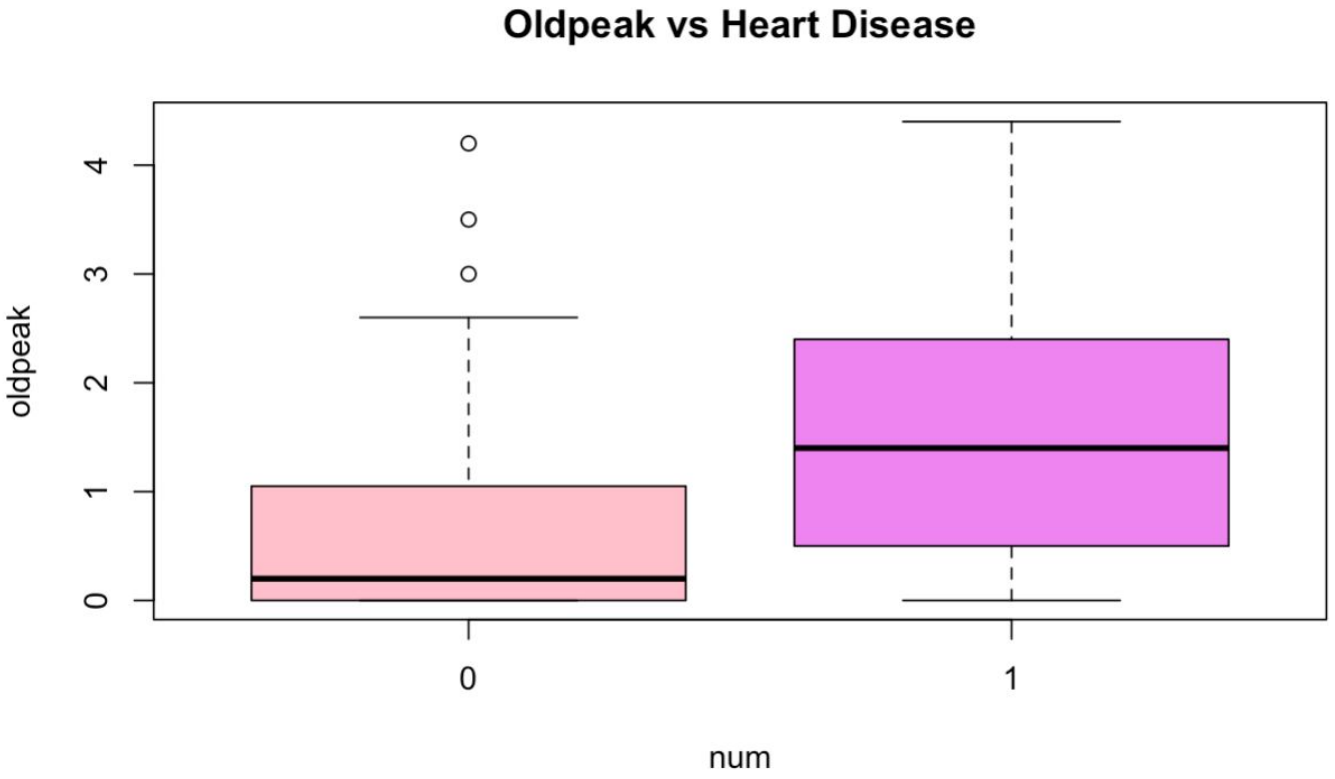
Exploratory Data Analysis: Multivariate Analysis



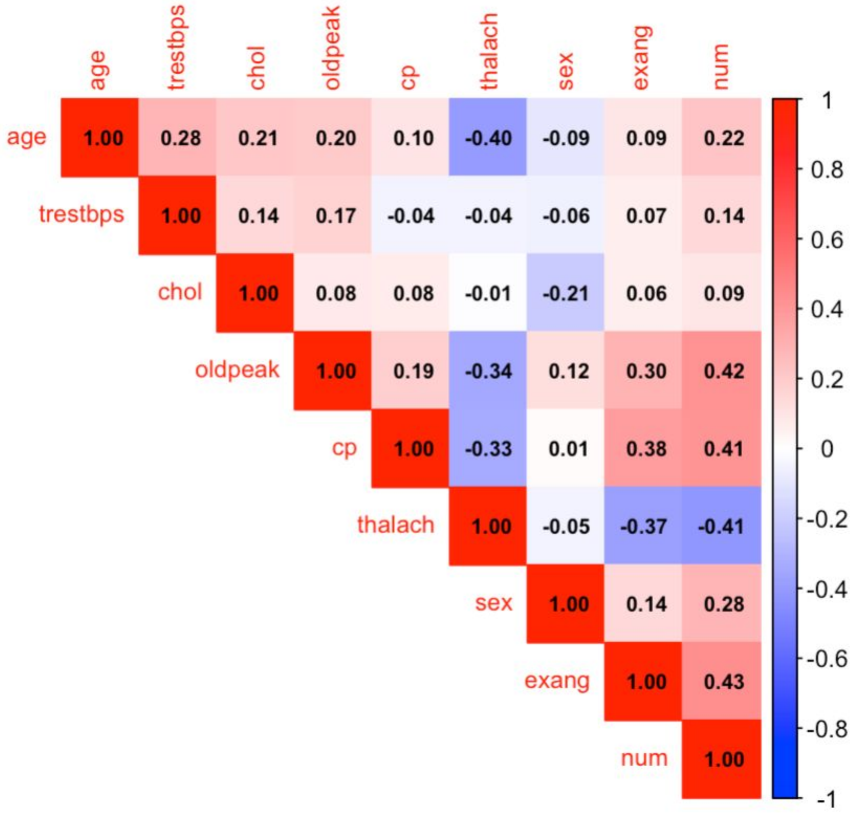
Exploratory Data Analysis: Multivariate Analysis



Exploratory Data Analysis: Multivariate Analysis



Exploratory Data Analysis: Multivariate Analysis



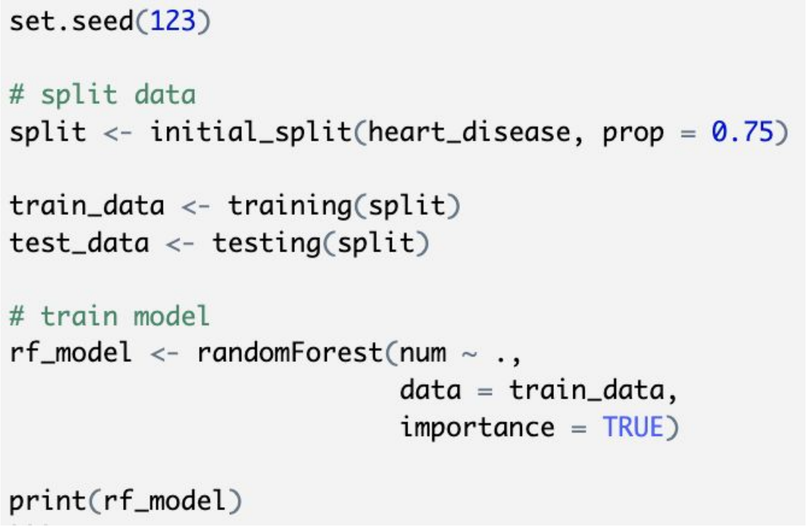
	age	trestbps	chol	oldpeak	cp	thalach	sex	exang	num
age	1.00	0.28	0.21	0.20	0.10	-0.40	-0.09	0.09	0.22
trestbps		1.00	0.14	0.17	-0.04	-0.04	-0.06	0.07	0.14
chol			1.00	0.08	0.08	-0.01	-0.21	0.06	0.09
oldpeak				1.00	0.19	-0.34	0.12	0.30	0.42
cp					1.00	-0.33	0.01	0.38	0.41
thalach						1.00	-0.05	-0.37	-0.41
sex							1.00	0.14	0.28
exang								1.00	0.43
num									1.00

- oldpeak and num have a moderate positive correlation (0.42)
- exang and num also a moderate positive correlation (0.43)
- weak correlation between sex and num (0.28)
- weak negative correlation between chol and sex

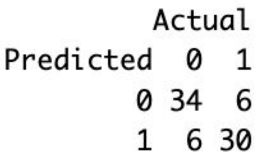
Random Forest Model

- **Binary Classification:** Random forests are well-suited for binary classification tasks and are more robust and accurate compared to simpler models like logistic regression.
- **Complexity:** In my case, logistic regression either overfitted the data and memorized the training data, or it underfitted, failing to capture underlying patterns in the variables because of its linear assumptions.
- **Non-linearity:** Handling non-linear relationships between variables is important in this case (and in any medical context) where interactions between multiple features influence the outcome.

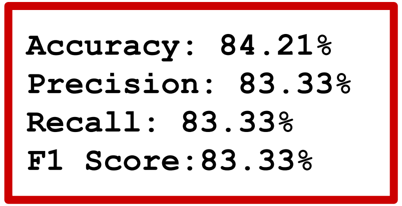
Random Forest Model



```
set.seed(123)

# split data
split <- initial_split(heart_disease, prop = 0.75)

train_data <- training(split)
test_data <- testing(split)

# train model
rf_model <- randomForest(num ~ .,
                         data = train_data,
                         importance = TRUE)


print(rf_model)
```



```
         Actual
Predicted  0  1
        0 34  6
        1  6 30
```



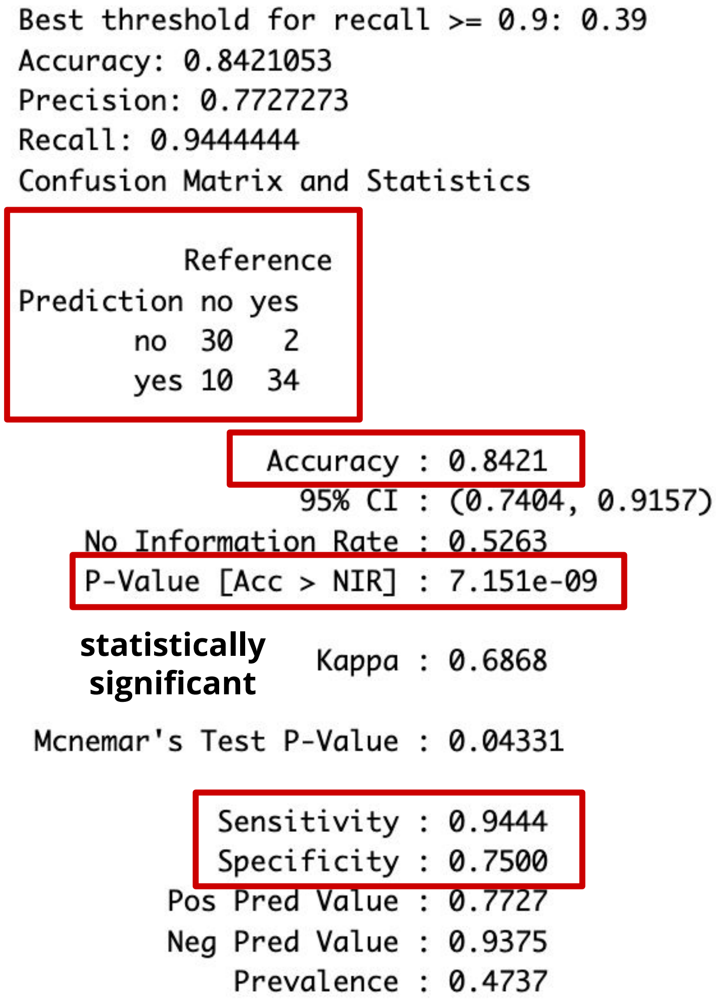
```
Accuracy: 84.21%
Precision: 83.33%
Recall: 83.33%
F1 Score:83.33%
```

My model’s recall of 0.8333 suggests that the model identified the majority of true positives correctly. However, increasing this is important, because in a medical context, it is less dangerous to have false positives than a false negative. Its precision of 0.8333 indicates that the model is moderately trustworthy (when it predicts someone has heart disease, there is an ~83% that it is right). The F1 Score 0.8333 shows a good balance between precision and recall.

Improving the Model

To improve my model (with a focus on recall), I used cross-validation and grid search to let the model choose optimal thresholds and hyperparameters based on specific metrics (in this case, I used recall and accuracy).

Final Model



```
Best threshold for recall >= 0.9: 0.39
Accuracy: 0.8421053
Precision: 0.7727273
Recall: 0.9444444
Confusion Matrix and Statistics

          Reference
Prediction no yes
       no  30   2
       yes 10  34

               Accuracy : 0.8421
                 95% CI : (0.7404, 0.9157)
    No Information Rate : 0.5263
    P-Value [Acc > NIR] : 7.151e-09

                  Kappa : 0.6868

 Mcnemar's Test P-Value : 0.04331

            Sensitivity : 0.9444
            Specificity : 0.7500
         Pos Pred Value : 0.7727
         Neg Pred Value : 0.9375
             Prevalence : 0.4737
```

statistically significant

I traded off precision and accuracy for recall. This is because in the real world, medical data is often imbalanced (few people have the disease), so the model could get a high accuracy just by predicting “no disease” all the time. High precision means fewer false positives – not wrongly alarming healthy people.

Having a high recall means the model is catching almost all the sick patients even if it wrongly predicts healthy people have the disease, which is useful for early diagnosis or intervention. The f1 score reflects a healthy trade-off between the two.

Conclusion

Through this project, I confirmed that patients that are older, male, have a lower maximum heart rate, and have ST depression, are more likely to have heart disease. Surprisingly, cholesterol levels were not always a strong indicator of heart disease and had to be interpreted with other features.

To maximize recall, my model used a threshold of 0.39 instead of 0.5, showing how it was struggling a bit with predicting true positives. This makes sense, as the negative class (1) was the minority class of my dataset.

Overall, my model can be used in the real world as a screening tool to prioritize patients for further tests, ultimately contributing to improved patient outcomes through early detection.

Future Work

My model traded off precision and accuracy for recall. However, in the real world, this can also become expensive, as hospitals would run tests for people who in fact do not have heart disease. To reduce this trade off, I could use a model such as XGBoost, which is scalable, fast, and can handle various tasks like regression, classification, ranking, and regularization, which is important for small datasets such as this one.

I could also upsample the negative class (1) of my dataset, as it was slightly underrepresented compared to the positive class (0).

I could also try to implement my model on the other two datasets from the folder (Hungary and Switzerland), to ensure that my model is robust, is applicable to different populations, and was not affected by data leakage during this initial project.

Thank you!